Fumihiko Sano and Kouichi Sakurai "DES compatible 128-bits key block cipher DES-SS"

Lines 3-10 of the Right Column of the First Page

This research proposes an improved DES-SS that uses a larger number of key bits than that of the conventional DES. Although the proposed system cannot use a conventional DES chip as it is, the same function is easily attained on a software level Unlike Triple-DES, the proposed system does not undergo a decrease in the processing speed Further, the conventional differential decoding and the conventional linear decoding are not applicable. Still further, the proposed DES-SS is compatible with the conventional DES as long as the keys it uses satisfy the predetermined conditions.

DESとの互換性を考慮した128ビット鍵長ブロック暗号 DES-SS
DES compatible 128-bits key block cipher DES-SS

佐野 文彦* 櫻井 幸一*
Fumihiko Sano Kouichi Sakurai

Abstract— Data Encryption Standard(DES) is a 64 bits key length block cipher. The most pertinent criticism of DES is the length of the key, 64 bits is said to be too small to protect against a key exhaustive search. Some improvements are supposed(e.g. DESX, Triple-DES), but they are not same data structure as DES. We suppose a new DES-like block cipher DES-SS which has 128-bits key. DES-SS can be used as DES compatible when the key satisfies a certain condition.

Keywords— DES, secret key cipher, key schedule

1 はじめに

データ暗号化規格 (Data Encryption Standard)[1] は，世界で最も広く用いられている秘密鍵暗号系である．1977年に発表されて以来，DESに対してはさまざまな観点からその安全性について評価が行われている．その結果，1990年ごろに差分解読法 [2] や線形解読法 [3] といった鍵の全数探索よりも効率的な解読法が次々と提案された．特に線形解読法では，標準である16段DESの解読に成功している．

DESで特に問題となるのは鍵の短さである．開発された当初から実質56ビットの鍵では短すぎるとの指摘があったが，ハードウェア技術の発達により，現在では 2^{56} 個の鍵の組み合わせを調べるハードウェアの実現が可能であることが知られている．DESの鍵長の問題に対しては，DESチップをそのまま使用しながら，鍵を長くする方法もいくつか (e.g. DESX, Triple-DES) 提案されている [4]

DESXは，DESの入力，出力にそれぞれ64ビットの鍵をx-orするだけの簡単な構造であるが，鍵の全数探索に対しては理論的な安全性が証明されている [5]．しかし，この方法では，差分解読法 [2] や線形解読法 [3] に対しては，DESと同等の安全性しか達成されない．これに対し，Triple-DESは，DESを三重に処理するアルゴリズムなので，鍵の全数探索だけでなく，差分解読法や線形解読法に対しても，DES以上の安全性が確保できると考えられている．だが，Triple-DESは，当然，処理速度がDESの1/3弱に落ちる．

本研究では，DESの鍵ビットを増加させる改良DES-SSを提案する．提案方式では，DESチップをそのまま流用することはできないが，ソフトウエアレベルでは，容易に実装可能である．また，処理速度もTriple-DESほど劣化しない．加えて，従来の差分解読法や線形解読法がそのまま適用できない構造となっている．さらに，このDES-SSは，鍵が一定の条件を満たせば本来のDESとの互換性を実現することも可能である．

2 DESの問題点

DESは64ビット入力64ビット出力のブロック暗号である．鍵は64ビット長であるが，8ビットは鍵パリティであり用いられないため，実質56ビット鍵の暗号である．DESは以下のような問題点を抱えている．

1. 64ビットブロック暗号である
   現在のハードウェア技術では128ビットブロック暗号を構成した方がより高速に暗号化が可能である．
2. F関数の構造の問題
   DESが開発された当時では，線形解読法がまだ発案されておらず，線形解読法に対する耐性が考慮されていない．また，ハードウェア技術の制約からF関数に含まれるS-boxのテーブルサイズが小さい．現在では，ハードウェア技術の発達により，S-boxのテーブルサイズをより大きくして安全性を高めることが可能であり，望ましいと考えられている．
3. 鍵の長さが56ビットである
   DESは開発当初から，56ビットの鍵の長さでは安全でないとの懸念がなされていたが，近年，専用ハードウェアを用いることにより，鍵の全数探索が可能であるとの結果が発表されている．また，DESの解読方法の一つである線形解読法は，鍵の一部ビットを解読により特定して，残りのビットに対して全数探索を行う．このため，鍵を増やすことにより残り鍵ビットの全数探索を行う手間が増大し安全性が向上すると考えられる．

* 九州大学大学院システム情報科学研究科，〒812-81 福岡市東区箱崎6-10-1 Department of Computer Science and Communication Engineering Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-81, Japan
E-mail:sano@csce.kyushu-u.ac.jp,sakurai@csce.kyushu-u.ac.jp

図 1: 鍵スケジュール部の構造

1,2 番目の問題点は DES の構造自体にかかわっており改良は容易でない．また，改良を行えば，本来の DES とは異なった暗号化処理を行うことになる．したがって，DES で暗号化されたデータを復号化できないといった，互換性の問題が発生する．本研究では，特に 3 番目の鍵の長さの問題点に着目し，鍵の長さを 128 ビットに増加させるとともに，DES との互換性を持った改良暗号 DES-SS を提案する．

3 DES の改良

DES-SS では，DES の鍵ビットのサイズの問題点に対して改良を試みる．本稿で提案する DES-SS では，DES では 64 ビットであった鍵が 128 ビットに拡張され，実行鍵ビット長は 56 ビットから二倍の 112 ビットに増加する．

1. 鍵スケジュール部の構成
   鍵スケジュール部の構造は図 1 で表される．DES との互換性を維持するために，128 ビットの鍵を 64 ビットずつに分割し，それぞれに DES の鍵スケジュール部 (S_A,S_B) を適用する．F 関数に用いる拡大鍵は S_A の生成する拡大鍵をそのまま用いる．F2 関数に用いられる鍵は S_A と S_B のビット毎の排他的論理和を 16 ビットずつのブロックに分割し，それぞれ，G1,G2,G3 として用いる．また，拡大鍵を生成するビット選択部で，56 ビットの左から {9,18,22,25,35} 番目の 5 ビットを選択し，F2 関数のシフト鍵として用いる
2. 使用鍵ビットの増大
   DES の各段で使用される鍵のビット数を増加させるために，各段の入力側に F2 関数 (図 3) を組み込み，一段の構造を図 2 で表される構造にする．
3. F2 関数
   F2 関数の構造は図 3 で表される．32 ビットの入力を 16 ビットずつに分割し，図のように鍵とのビットごとの論理和や論理積をとる．巡回シフト部は 32 ビットのデータに対して適用され，シフト鍵の 5 ビットを 2 進表記とみなし，そのビット数だけ巡回シフトを行う．

図 2: 各段の構造

図 3 鍵挿入部 F2 の構造

この方法の利点は，DES と同一の鍵スケジュールを用いていることにある．64 ビットの鍵を複製して 128 ビットの鍵を作成することにより，DES との完全な互換性を持つことが可能となる．なぜなら，二つの鍵スケジュール部の入力が同じならば，各段で生成される拡大鍵も同一のものとなる．両者のビット毎の排他的論理和は 0 となるため，F2 関数に使用される G1,G2,G3 は 0，シフト鍵は 16 となる．この場合，F2 関数の構造は無視されるので，128 ビット鍵の前半 64 ビットを鍵として用いた

DESの場合と全く同じ処理となり，DESで暗号化されたデータの復号化(暗号化処理)が可能であり，互換性が満足される．

F2関数で用いられる操作は，巡回シフト命令，論理積，排他的論理和である．これらの演算は多くのハードウェアで一般的に実装されている命令であり，高速な操作が可能である

また，DES-SSはF2関数での鍵の加え方に工夫を行っている．鍵を加える場合，データと鍵の排他的論理和を取る方法もある．しかし，この方法では線形解読法を適用した場合，排他的論理和で加えられた鍵は解読の過程で比較的容易に導出されてしまい，鍵のビットを増やしても効果が少ない．それに対して，論理積を用いて鍵を加えた場合，線形解読法による鍵の直接の導出は行われないので，鍵ビットの増加により強度の向上が計れると考える．

4 速度

実際にDES-SSのプログラムをC言語を用いて作成し，ECBモードで暗号化した場合でのスループットを計測した．速度比の参考として，同じソースから，DES-SSのF2関数に関する部分を削除して計測したものをDESのスループットとして示す．ベンチマークテストの方法は[6]を参考にした．

図 4: スループット

OS	: Linux
CPU	: Pentium 120MHz
コンパイラ	: gcc-2.7.2.1
最適化オプション	: -O2
暗号化モード	ECBモード

実験結果では，2^{10}ブロック(8KB)以上のデータを暗号化する場合，鍵生成部の処理の影響が小さくなる．2^{10}ブロック以上のデータ量の場合，DES-SSのスループットはDESの65%程度の処理速度である．DESを三重に処理するTriple-DESがDESの1/3程度の速度であるのに対して高速である．

5 おわりに

本稿で提案したDESの改良では，F2関数により各段でデータの処理が増加する．F2関数は，論理演算やシフト演算といった一般的に計算機に実装されている演算を用いており，高速な処理が可能であるので効率はあまり低下しないと考る．また，F2関数により線形解読法などによる攻撃に強くなると考えられるが，鍵の依存性の問題もある．また，F2関数を組み込むことによる，差分解読法，線形解読法に対する強度の変化についてのさらなる評価が必要である．

参考文献

[1] "Data Encryption Standard," Federal Information Processing Standards Publicatom 46,National Bureau of Standards,U.S Department of Commerce, (1977).

[2] E.Biham and A.Shamir,"Differential Cryptanalysis of DES-like Cryptosystems," Journal of CRYPTOLOGY,Vol.4,Number 1,1991

[3] 松井 充, "DES暗号の線形解読 (I)," 暗号と情報セキュリティシンポジウム, SCIS93-3C, (1993).

[4] B.Schneier, "Applied Cryptography," 2nd edition, Wiley (1996).

[5] J.Kilian and P.Rogaway, "How to protect DES against exhaustive key search," Proc CRYPTO'96, pp.252-267 (1996).

[6] 酒井 康行, 松井 充, "秘密鍵暗号ベンチマークプログラム設計に関する一考察," 信学技報, ISEC96-28, (1996)

Jpn Pat. Appln. KOKAI Publication No. 10-116029

Filing No.: 8-269897
Filing Date: October 11, 1996
Applicant: Kabushiki Kaisha Toshiba
KOKAI Date: May 6, 1998
Request for Examination: Not filed
Int.Cl G09C 1/00

---

(57) [Abstract]

[Object] The object is to provide an encrypting apparatus which provides a high degree of security while maintaining the compatibility with DES.

[Means for Achieving the Object] The encrypting apparatus comprises two key schedule sections A and B which are identical in configuration and which develop two encrypting keys, obtained by equally dividing key information of a predetermined bit sequence, into intermediate keys used for agitating an input message; an XOR section 14 which outputs an exclusive OR with respect to the two intermediate keys output from the two key schedule sections A and B; and an agitating section which agitates the input message by using one of the intermediate keys when the exclusive OR is "0" and therefore indicates that the two intermediate keys are the same, and which agitates the input message on the basis of the two intermediate keys when the two intermediate keys are different.

(19)日本国特許庁（ＪＰ）　(12)**公開特許公報（Ａ）**　(11)特許出願公開番号

特開平10－116029

(43)公開日　平成10年(1998)5月6日

(51)Int.Cl.[6]	識別記号	ＦＩ	
Ｇ０９Ｃ　1/00	６１０	Ｇ０９Ｃ　1/00	６１０Ａ
			６１０Ｂ

審査請求　未請求　請求項の数4　ＯＬ　（全７頁）

(21)出願番号　特願平8－269897

(22)出願日　平成８年(1996)10月11日

(71)出願人　000003078
株式会社東芝
神奈川県川崎市幸区堀川町72番地

(72)発明者　佐野　文彦
福岡県福岡市東区西松３－７－25　木栄荘201号

(72)発明者　櫻井　幸一
福岡県福岡市城南区七隈２－16－22

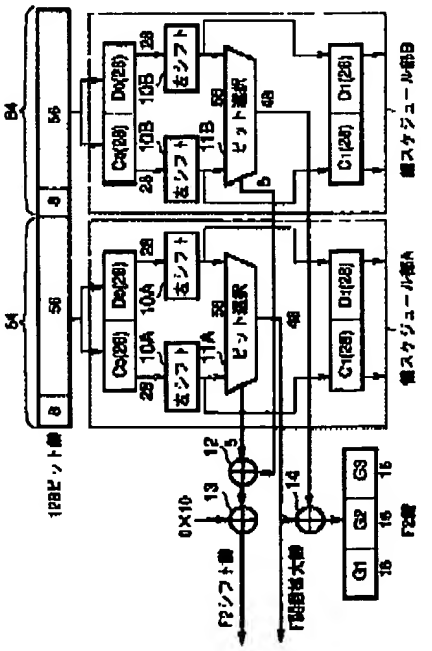
(74)代理人　弁理士　鈴江　武彦　（外６名）

(54)【発明の名称】　暗号化装置及び暗号化方法

(57)【要約】

【課題】ＤＥＳとの互換性を維持しつつ安全性を増大することができる暗号化装置を提供する。

【解決手段】所定のビット列からなる鍵情報を等分して得られる２つの暗号化鍵を、入力電文を攪拌するのに用いられる中間鍵にそれぞれ展開する同一構成の２つの鍵スケジュール部Ａ、Ｂと、この２つの鍵スケジュール部Ａ、Ｂから出力された２つの中間鍵に対して排他的論理和を求める排他的論理和１４と、この排他的論理和が０となることにより２つの中間鍵が互いに同一であることが検出された場合には、いずれか１つの中間鍵を用いて入力電文を攪拌するとともに、比較された２つの中間鍵が互いに同一でないことが検出された場合には、２つの中間鍵に基づいて入力電文を攪拌する攪拌部とを具備する。



1

【特許請求の範囲】

【請求項１】　入力電文を外部から入力された鍵情報に依存して攪拌し，対応する符号化電文を出力するデータ攪拌部と，

前記鍵情報を前記データ攪拌部に供給される中間鍵に展開する鍵スケジュール部とからなる暗号化装置であって，

前記鍵スケジュール部は，

鍵情報の半数のビットを一巡の出力に対応づける鍵展開手段と，

外部から入力された前記鍵ビットの内，半数を前記鍵展開手段にて第１の中間鍵に展開すると共に，全鍵ビットの残りの半数に同じ処理を施して第１の中間鍵と同数のビットからなる第２の中間鍵に展開し，該第２の中間鍵と第１の中間鍵のビット毎に所定の演算を行なうことにより，第３の中間鍵を得る手段とを有し，

前記データ攪拌部は，

前記第１乃至第３の中間鍵の一部または全部のビットの値によって規定される攪拌処理を実現する複数の攪拌手段を有し，

前記複数の攪拌手段の内，第３の中間鍵のみによって規定される攪拌手段は，第３の中間鍵のビットの内，前記攪拌手段で用いられているビットが特定の条件を満たした場合には，前記攪拌手段への入力ビット列と同一のビット列を出力するように構成されていることを特徴とする暗号化装置。

【請求項２】　所定のビット列からなる鍵情報を複数個に分割して得られる複数の暗号化鍵を，入力電文を攪拌するのに用いられる中間鍵にそれぞれ展開する同一構成の複数の鍵展開手段と，

この複数の鍵展開手段から出力された複数の中間鍵を互いに比較する比較手段と，

この比較手段によって比較された複数の中間鍵が同一であることが検出された場合には，複数の中間鍵のいずれか１つを用いて入力電文を攪拌するとともに，

比較された複数の中間鍵が同一でないことが検出された場合には，複数の中間鍵のすべてに基づいて入力電文を攪拌する攪拌手段と，

を具備することを特徴とする暗号化装置。

【請求項３】　外部から入力された鍵情報を鍵スケジュール部において中間鍵に展開し，データ攪拌部においてこの中間鍵に依存して入力電文を攪拌して対応する符号化電文を出力する暗号化方法であって，

所定の鍵展開手段によって，外部から入力された鍵情報の内，半数のビットを第１の中間鍵に展開すると共に，前記鍵情報の残りのビットに同じ処理を施して第１の中間鍵と同数のビットからなる第２の中間鍵に展開し，

さらに，第１の中間鍵と第２の中間鍵との間でビット毎に所定の演算を行なうことにより，第３の中間鍵を得るとともに，

2

前記第１，第２，第３の中間鍵の一部または全部のビットの値によって規定される攪拌処理を行なうにあたって，第３の中間鍵のみによって規定される攪拌処理は，第３の中間鍵が特定の条件を満たした場合には，入力ビット列と同一のビット列を出力することを特徴とする暗号化方法。

【請求項４】　同一構成の複数の鍵展開手段によって，所定のビット列からなる鍵情報を複数個に分割して得られる複数の暗号化鍵を入力電文を攪拌するのに用いられる中間鍵にそれぞれ展開する展開工程と，

前記複数の鍵展開手段から出力された複数の中間鍵を互いに比較する比較工程と，

この比較工程において比較された複数の中間鍵が同一であることが検出された場合には，複数の中間鍵のいずれか１つを用いて入力電文を攪拌するとともに，

比較された複数の中間鍵が同一でないことが検出された場合には，複数の中間鍵のすべてに基づいて入力電文を攪拌する攪拌工程と，

を具備することを特徴とする暗号化方法。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は暗号化装置及び暗号化方法に関し，特に，秘密鍵ブロック暗号を用いた暗号化装置及び暗号化方法に関する。

【０００２】

【従来の技術】ＤＥＳ（Data Encryption Standard）は，現在，最も広範に用いられている秘密鍵ブロック暗号系であり，文献，"Data Encryption Standard, "Federal Information Processing Standards Publication 46, National Bureau of Standards, U.S. Department of Commerce, 1977. に詳細に記載されている。

【０００３】ＤＥＳは６４ビット入力，６４ビット出力のブロック暗号であり，そのうち８ビットは鍵パリティ用として使用されるので５６ビットのみが実質的な鍵である。したがって，ＤＥＳが開発された当初からその安全性について議論がなされており，１９７７年に発表されて以来，さまざまな観点からの評価が行われている。その結果，１９９０年ごろに差分解読法や線形解読法といった鍵の全数探索よりも効率的な解読法が次々と提案された。特に，線形解読法を用いることにより，標準である１６段ＤＥＳの解読に成功している。

【０００４】なお，差分解読法については，文献，E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems," Journal of CRYPTOLOGY, Vol 4, Number 1, 1991に，線形解読法については，文献，松井充，"ＤＥＳ暗号の線形解読（Ｉ）"，暗号と情報セキュリティシンポジウム，ＳＣＩＳ９３－３Ｃ，１９９３に記載されている。

【０００５】ＤＥＳが開発されてからの技術の発展や上記したような解読法を考慮すると，ＤＥＳには以下のよ

うな問題点があると考えられる。

（1）　現在のハードウェア技術の進歩を考えると処理を高速に行なうために１２８ビットのブロック暗号を構成することが可能であるが、いぜんとして６４ビットのブロック暗号を用いている。

（2）　F関数の構造に問題がある。ＤＥＳが開発された当時では、線形解読法が発案されておらず、線形解読法に対する耐性が考慮されていない。また、ハードウェア技術の発達の制約からF関数に含まれるS箱のテーブルサイズが小さい。現在のハードウェア技術のレベルを考えるとS箱のサイズをより大きくして安全性を高めることが望ましい。

（3）　５６ビットからなる鍵を用いているが、５６ビットの鍵の長さでは安全ではない。ハードウェアの発達により、現在では2^{56}個の鍵の組合せを調べるハードウェアの実現が可能である。また、専用のハードウェアを用いることにより、鍵の全数探索が可能であるとの発表もなされている。また、ＤＥＳの解読方法の１つである線形解読法は、鍵の一部のビットを解読により特定して残りのビットに対して全数検索を行なうものであるから、鍵を増やすことにより残りの鍵ビットの全数検索を行なう手間が増大して安全性が向上すると考えられる。

【０００６】上記した問題において、（3）のＤＥＳの鍵の長さに関する問題については、ＤＥＳチップをそのまま使用しながら鍵を長くする方法が考えられている。例えば、論文、B.Schneier,"Applied Cryptography," 2nd edition, Wiley (1996)は、ＤＥＳＸやTripleＤＥＳなどの暗号化方法を開示している。

【０００７】ＤＥＳＸはＤＥＳの入力及び出力の各々と６４ビットの鍵との排他的論理和をとる方法であるが、鍵の全数探索に対しては理論的な安全性が証明されている（J.Kilian and P.Rogaway, "How to protect ＤＥＳ against exhaustive key search," Proc. CRYPTO'96, pp.252-267(1996) を参照）。また、TripleＤＥＳはＤＥＳを３重に処理するアルゴリズムを用いているので、鍵の全数探索だけでなく、差分解読法や線形解読法に対しても、ＤＥＳ以上の安全性が確保できる。

【０００８】

【発明が解決しようとする課題】しかしながら、上記したＤＥＳＸは差分解読法や線形解読法に対してはＤＥＳと同等の安全性しか達成されない。また、TripleＤＥＳは構成が３倍になるので、処理速度はＤＥＳの処理速度の１／３となる。

【０００９】一方、上記した（1）、（2）で述べた問題点はＤＥＳの構造自体に関しており、これらを改良することは容易でない。また、これらを改良すれば本体のＤＥＳとは異なった暗号化処理を行なうことになり、ＤＥＳとの互換性の問題が新たに発生してしまう。

【００１０】本発明の暗号化装置及び暗号化方法はこのような課題に着目してなされたものであり、その目的とするところは、ＤＥＳとの互換性を維持しつつ安全性を増大することができる暗号化装置及び暗号化方法を提供することにある。

【００１１】

【課題を解決するための手段】上記の目的を達成するために、第１の発明に係る暗号化装置は、入力電文を外部から入力された鍵情報に依存して攪拌し、対応する符号化電文を出力するデータ攪拌部と、前記鍵情報を前記データ攪拌部に供給される中間鍵に展開する鍵スケジュール部とからなる暗号化装置であって、前記鍵スケジュール部は、鍵情報の半数のビットを一意の出力に対応づける鍵展開手段と、外部から入力された前記鍵ビットの内、半数を前記鍵展開手段にて第１の中間鍵に展開すると共に、全鍵ビットの残りの半数に同じ処理を施して第１の中間鍵と同数のビットからなる第２の中間鍵に展開し、該第２の中間鍵と第１の中間鍵のビット毎に所定の演算を行なうことにより、第３の中間鍵を得る手段とを有し、前記データ攪拌部は、前記第１乃至第３の中間鍵の一部または全部のビットの値によって規定される攪拌処理を実現する複数の攪拌手段を有し、前記複数の攪拌手段の内、第３の中間鍵のみによって規定される攪拌手段は、第３の中間鍵のビットの内、前記攪拌手段で用いられているビットが特定の条件を満たした場合には、前記攪拌手段への入力ビット列と同一のビット列を出力するように構成されている。

【００１２】また、第２の発明に係る暗号化装置は、所定のビット列からなる鍵情報を複数個に分割して得られる複数の暗号化鍵を入力電文を攪拌するのに用いられる中間鍵にそれぞれ展開する同一構成の複数の鍵展開手段と、この複数の鍵展開手段から出力された複数の中間鍵を互いに比較する比較手段と、この比較手段によって比較された複数の中間鍵が同一であることが検出された場合には、複数の中間鍵のいずれか１つを用いて入力電文を攪拌するとともに、比較された複数の中間鍵が同一でないことが検出された場合には、複数の中間鍵のすべてに基づいて入力電文を攪拌する攪拌手段とを具備する。

【００１３】また、第３の発明に係る暗号化方法は、外部から入力された鍵情報を鍵スケジュール部において中間鍵に展開し、データ攪拌部においてこの中間鍵に依存して入力電文を攪拌して対応する符号化電文を出力する暗号化方法であって、所定の鍵展開手段によって、外部から入力された鍵情報の内、半数のビットを第１の中間鍵に展開すると共に、前記鍵情報の残りのビットに同じ処理を施して第１の中間鍵と同数のビットからなる第２の中間鍵に展開し、さらに、第１の中間鍵と第２の中間鍵との間でビット毎に所定の演算を行なうことにより、第３の中間鍵を得るとともに、前記第１、第２、第３の中間鍵の一部または全部のビットの値によって規定される攪拌処理を行なうにあたって、第３の中間鍵のみによって規定される攪拌処理は、第３の中間鍵が特定の条件

5

を満たした場合には，入力ビット列と同一のビット列を出力する。

【００１４】また，第４の発明に係る暗号化方法は，同一構成の複数の鍵展開手段によって，所定のビット列からなる鍵情報を複数個に分割して得られる複数の暗号化鍵を入力電文を攪拌するのに用いられる中間鍵にそれぞれ展開する展開工程と，前記複数の鍵展開手段から出力された複数の中間鍵を互いに比較する比較工程と，この比較工程において比較された複数の中間鍵が同一であることが検出された場合には，複数の中間鍵のいずれか１つを用いて入力電文を攪拌するとともに，比較された複数の中間鍵が同一でないことが検出された場合には，複数の中間鍵のすべてに基づいて入力電文を攪拌する攪拌工程とを具備する。

【００１５】

【発明の実施の形態】以下，図面を参照して本発明の一実施形態を詳細に説明する。図１は本実施形態に係る暗号化装置の構成を示す図であり，入力電文としての平文（６４ビット）を外部から入力された鍵情報ｋに依存して攪拌し，対応する符号化電文を出力する第１段～第１６段から構成されるデータ攪拌部と，鍵情報ｋを前記データ攪拌部に供給される中間鍵に展開する鍵スケジュール部４とからなる。

【００１６】図１において，平文（６４ビット）は初期転置ＩＰが施された後，２つに等分されて左側３２ビットL$_0$と右側３２ビットR$_0$が生成される。一方，鍵スケジュール４には１２８ビットの鍵情報Ｋが入力される。鍵スケジュール部４は以下に述べる方法によってＦ関数拡大鍵とＦ２鍵及びＦ２シフト鍵を生成して，データ攪拌部のＦ関数２とＦ２関数３にそれぞれ入力する。ここでＦ関数２は通常のＤＥＳと同様の攪拌処理を行なうものであり，Ｆ２関数３は以下に述べるような攪拌処理を行なう。

【００１７】Ｆ関数２はＦ関数拡大鍵とＦ２関数３の出力とを受けて所定の攪拌処理を行ってその結果を排他的論理和１に入力する。排他的論理和１はL$_0$（３２ビット）とＦ関数２の出力との間の排他的論理和を出力するが，これによって次段の右側３２ビットR$_1$が得られる。また，Ｆ２関数３の出力は次段の左側３２ビットL$_1$となる。

【００１８】以上の攪拌処理が第１段で行われ，L$_1$（３２ビット）とR$_1$（３２ビット）とが第２段に送られて第１段と同様の処理が施される。このようにして第１６段までの攪拌処理が行われた後，最終転置ＩＰ$^{-1}$が施されて暗号文（６４ビット）が得られる。

【００１９】このように本実施形態では，ＤＥＳの各段で使用される鍵のビット数を増加させるために，通常のＤＥＳの構成に加えて，各段の入力側（図１に示す位置）にＦ２関数を組み込んでいる。

【００２０】図２は図１に示す鍵スケジュール部４の１

6

段分の構成を示しており，ＤＥＳと同一の構成の鍵スケジュール部Ａと鍵スケジュール部Ｂとからなる。したがって鍵スケジュール部４はこのような構成の鍵スケジュール部を１６段設けた構成を有する。鍵スケジュール部Ａ，Ｂとも同一の処理を行なうのでここでは鍵スケジュール部Ａについてのみ説明する。

【００２１】１２８ビットの鍵情報は半分ずつに分割され，縮約型転置ＰＣ－１を施された後，５６ビットの鍵として各々の鍵スケジュール部に入力される。鍵（５６ビット）を２つに等分して生成した２８ビットからなる２つの鍵の各々C$_0$（２８ビット），D$_0$（２８ビット）について左シフト部１０Ａで左シフト処理を施した後，ビット選択部１１Ａに入力する。ビット選択部１１Ａは所定のビット選択処理により４８ビットからなる第１の中間鍵と，５ビットからなる鍵とを出力する。この５ビットの鍵としては，５６ビット鍵の例えば，左から９，１８，２２，２５，３５番目の５ビットが用いられる。同様にしてビット選択部１０Ｂからは４８ビットからなる第２の中間鍵と，５ビットからなる鍵とが出力される。

【００２２】そして，ビット選択部１１Ａからの４８ビットの鍵とビット選択部１１Ｂからの４８ビットの鍵との間で排他的論理和１４をとり，その結果としての４８ビットの第３の中間鍵を３等分して１６ビットの鍵C１，Ｇ２，Ｇ３を得る。この鍵Ｇ１，Ｇ２，Ｇ３をＦ２鍵としてＦ２関数３に入力する。

【００２３】同様に，鍵スケジュール部Ａのビット選択部１１Ａからの５ビットの鍵と，鍵スケジュール部Ｂのビット選択部１１Ｂからの５ビットの鍵との間で排他的論理和１２を求め，その結果と０Ｘ１０（０Ｘは１６進を表す）との間で排他的論理和１３をとったものをＦ２シフト鍵としてＦ２関数３に入力する。

【００２４】さらに，本実施形態では，鍵スケジュール部Ａのビット選択部１１Ａからの４８ビットの鍵をＦ関数拡大鍵としてＦ関数２に入力する。また，各々の２８ビットの鍵C$_0$（２８ビット），D$_0$（２８ビット）を左シフトすることによって得られたC$_1$（２８ビット），D$_1$（２８ビット）は次の段の鍵スケジュール部の入力となる。

【００２５】このように本実施形態では，ＤＥＳとの互換性を維持するために，１２８ビットの鍵を６４ビットずつに分割し，それぞれに対してＤＥＳの鍵スケジュール部Ａ，Ｂを適用している。また，Ｆ関数に入力される拡大鍵としては鍵スケジュール部Ａからの４８ビットの鍵をそのまま用いている。また，Ｆ２関数に用いる鍵としては鍵スケジュール部Ａからの４８ビット鍵と鍵スケジュール部Ｂからの４８ビット鍵とのビットごとの排他的論理和を１６ビットごとのブロックに分割して，それぞれＧ１，Ｇ２，Ｇ３としている。また，５６ビット鍵の左から特定番目の５ビットを選択してＦ２関数のシフ

7

ト鍵として用いている。

【００２６】図３はＦ２関数３の構成を示す図である。ここでは図１に示すＲｏ（３２ビット）を２つに等分して２つの１６ビットのブロックを生成する。左側の１６ビットブロックは排他的論理和２０に入力される。また，右側の１６ビットブロックは論理積２１と排他的論理和２２に入力される。

【００２７】論理積２１は１６ビット鍵Ｇ１と右側の１６ビット鍵との論理積をとって排他的論理和２０に入力する。排他的論理和２０は左側の１６ビットブロックと論理積２１からの鍵との間で排他的論理をとり，その結果を左巡回シフト部２３に入力する。

【００２８】一方，排他的論理和２２は１６ビット鍵Ｇ２と右側の１６ビットブロックとの間で排他的論理和をとり，その結果を左巡回シフト部２３に入力する。左巡回シフト部２３は，排他的論理和２０の出力と排他的論理和２２の出力からなる３２ビットの鍵に対して，入力された５ビットのＦ２シフト鍵を２進表記とみなして，そのビット数分の左巡回シフトを行なう。

【００２９】左巡回シフトを行った後の３２ビットの中間データの左半分の１６ビットはＦ２関数３の右半分の出力となる。また，左巡回シフトを行った後の３２ビットの中間データの右半分の１６ビットは排他的論理和２４に入力される。

【００３０】論理積２５は１６ビットの鍵Ｇ３と左巡回シフトを行った後の３２ビットの中間データの左半分の１６ビットとの論理積をとってその結果を排他的論理和２４に入力する。排他的論理和２４はこの論理積と左巡回シフトを行った後の３２ビットの中間データの右半分の１６ビットとの間の排他的論理和をとり，その結果をＦ２関数３の左半分として出力する。

【００３１】上記したように，本実施形態のＦ２関数では，３２ビットの入力を１６ビットずつに分割して，分割された鍵とのビットごとの論理和や論理積をとっている。また，巡回シフト部ではＦ２シフト鍵のビット数だけ３２ビットの鍵に対して巡回シフトを行っている。

【００３２】以上の説明からわかるように，本実施形態の暗号化装置の利点はＤＥＳと同一の鍵スケジュール構成を用いていることにある。６４ビットの鍵を複製して１２８ビットの鍵を生成することにより，ＤＥＳとの完全な互換性を維持することができる。なぜなら，２つの鍵スケジュール部Ａ，Ｂの６４ビットの入力が同じであるならば，各スケジュール部で生成される中間鍵も同一のものとなる。各スケジュール部で生成される中間鍵が同一ならば両者のビットごとの排他的論理和１４は０となるため，Ｆ２関数３で用いられるＧ１，Ｃ２，Ｃ３も０となる。また同様に，排他的論理和１２の出力も０となり，Ｆ２シフト鍵は０Ｘ１０となる。

【００３３】このとき，図３に示すＦ２関数３の排他的論理和２０，２２の一方の入力が０となるので，Ｆ２関数３に入力された２つの１６ビットの鍵はそのまま左巡回シフト部２３に入力される。さらに，左巡回シフト部２３では鍵スケジュール部Ａ，Ｂからの２つの５ビット鍵の間の排他的論理和１２と０Ｘ１０との排他的論理和１３をＦ２シフト鍵とし，このようなＦ２シフト鍵に基づいて左巡回を行っているので，Ｃ３＝０，すなわち，排他的論理和２４の一方の入力が０のときは，Ｆ２関数３に入力されたビット列（３２ビット）と同じビット列が出力されることになる。

8

【００３４】このようにして，２つの鍵スケジュール部Ａ，Ｂの６４ビットの入力が同じであるならば，Ｆ２関数３の構造は無視されることになるので，１２８ビット鍵の半分の６４ビットを鍵として用いたＤＥＳと全く同じ処理となり，互換性が満足される。

【００３５】さらに，本実施形態ではＦ２関数を加えたことによりＤＥＳと比較して各段でのデータ処理が増加することになるが，Ｆ２関数で用いられる演算は左巡回シフト命令，論理和，論理積，排他的論理和であり，これらの演算は通常多くのハードウェアに実装されているので高速処理が可能である。このことより，処理効率はＤＥＳと比較してあまり低下することはない。

【００３６】また，本実施形態ではＦ２関数での鍵の加え方に工夫を行っている。鍵を加える場合，入力データと鍵との排他的論理和をとる方法があるが，この方法では線形解読法を適用したときに排他的論理和で加えられた鍵は解読の過程で比較的容易に導出されてしまい，鍵のビット数を増やしても安全性を高める効果が少ない。これに対して，本実施形態では，論理和や論理積を用いて鍵を加えているので，線形解読法による鍵の直接の導出は行われず，鍵のビット数の増加により強度の安全性の向上が計れる。

【００３７】以下に，ＯＳとしてSolaris2.5，コンパイラとしてgcc-2.7.2を用い，最適化オプションなしのシステム環境下で，本実施形態の暗号化用サンプルソースプログラムを実行した場合のスループットを本実施形態とＤＥＳについて計測した。ここでは，上記サンプルソースプログラムからＦ２関数に関する部分を取り除いたものをＤＥＳの構成としている。

【００３８】計測の結果，図４（ａ）に示すような結果が得られた。スループットの速度比では本実施形態はＤＥＳの９０．６％であり，ＤＥＳを３重に構成したTripleＤＥＳがＤＥＳの１／３（すなわち，３３．３％）の速度であることと比較して十分高速であることがわかる。

【００３９】次に，ＯＳとしてLinux，ＣＰＵとしてPentium 120MHz，コンパイラとしてgcc-2.7.2.1，最適化オプションとして-O2を用いたシステム環境下で同様のサンプルソースプログラムを実行した場合のスループットを本実施形態とＤＥＳについて計測した。この場合も上記サンプルソースプログラムからＦ２関数に関する部

9　　10

分を取り除いたものをＤＥＳの構成としている。

【００４０】計測の結果、図４（ｂ）に示すような結果が得られた。図４（ｂ）によれば、スループットの速度比では本実施形態はＤＥＳの６５％であり、ＤＥＳを３重に構成したTripleＤＥＳがＤＥＳの１／３の速度であることと比較してまだ十分高速であることがわかる。

【００４１】

【発明の効果】本発明によれば、ＤＥＳとの互換性を維持しつつ安全性を増大することができる暗号化装置及び暗号化方法を提供できる。

【図面の簡単な説明】

【図１】本発明の一実施形態における暗号化装置の構成＊を示す図である。

【図２】図１に示す鍵スケジュール部４の１段分の構成を示す図である。

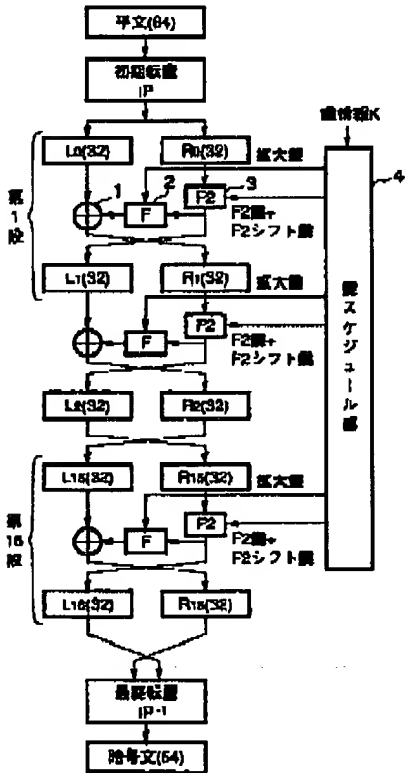
【図３】図１に示すＦ２関数の構成を示す図である。

【図４】本実施形態の暗号化方法に係るサンプルソースプログラムを実行した場合のスループットを本実施形態とＤＥＳについて計測した結果を示す図である。

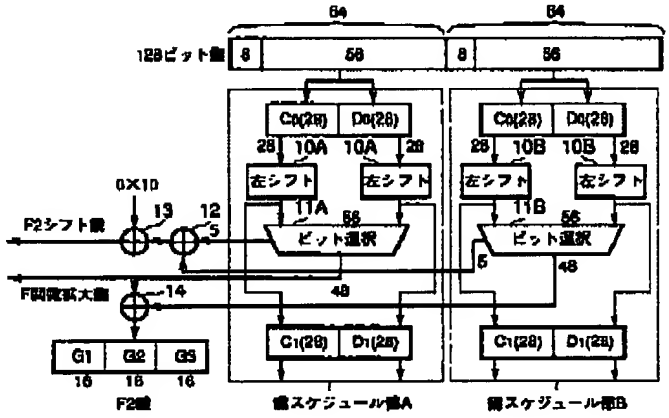
【符号の説明】

１…排他的論理和、２…Ｆ関数、３…Ｆ２関数、４…鍵スケジュール部、１０Ａ、１０Ｂ　左シフト部、１１Ａ、１１Ｂ　ビット選択部、１２、１３、１４　排他的論理和。

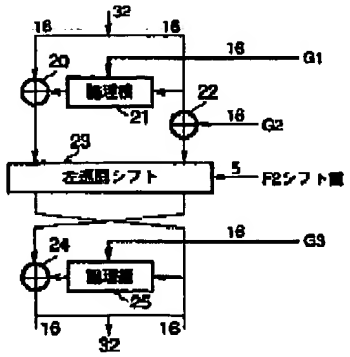
【図１】



【図２】



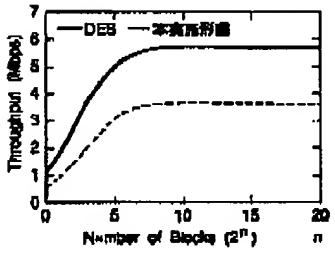
【図３】



【図４】

DES	本実施形態	速度比
64Kbps	58Kbps	90.6%

(a)



(b)